AD-751 124

VARIABLE ORDER FILTERS

Marvin Blum

Institute for Defense Analyses
Arlington, Virginia

December 1971

DISTRIBUTED BY:

NTIS

National Technical Information Service
U. S. DEPARTMENT OF COMMERCE
5285 Port Royal Road, Springfield Va. 22151

PAPER P-835

VARIABLE ORDER FILTERS

Marvin Blum

December 1971

IDA
INSTITUTE FOR DEFENSE ANALYSES
SCIENCE AND TECHNOLOGY DIVISION

IDA Log No. HQ 71-13827
Copy 25 of 30 copies

The work reported in this document was conducted under IDA's Independent Research Program. Its publication does not imply endorsement by the Department of Defense or any other government agency, nor should the contents be construed as reflecting the official position of any government agency.

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1 ORIGINATING ACTIVITY (Corporate author)
INSTITUTE FOR DEFENSE ANALYSES
400 Army-Navy Drive
Arlington, Virginia 22202

2a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

2b GROUP

3 REPORT TITLE

Variable Order Filters

4 DESCRIPTIVE NOTES (Type of report and inclusive dates)
Paper P-835, December 1971

5 AUTHOR(S) (First name, middle initial, last name)

Marvin Blum

6 REPORT DATE	7a. TOTAL NO OF PAGES	7b NO OF REFS
December 1971	33	12

8a CONTRACT OR GRANT NO
IDA Independent Research Program
b PROJECT NO
c
d

9a. ORIGINATOR'S REPORT NUMBER(S)
P-835

9b OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
None

10 DISTRIBUTION STATEMENT
This document is unclassified and suitable for public release.

11 SUPPLEMENTARY NOTES
N/A

12 SPONSORING MILITARY ACTIVITY
N/A

13 ABSTRACT

A variable order filter is characterized by a filter such that the dimension of the state vector varies with time. A special important case occurs for linear systems which can be generated from sampling a linear dynamic system, $\dot{x} = A(t)x + b(t)$, $y = M(t)x + \eta(t)$ where the term $b(t) = \Sigma b_j \delta_+(t - T_j)$, and $\delta_+(\tau)$ is the asymmetric Dirac function and the T_j are known as knots. Recursive algorithms are obtained given that T_j are known. Because of the simplicity of the algorithms, it is feasible to estimate both the magnitude of b_j and the knot position T_j by appending a detector to process the residuals or the innovation process. Using classic hypothesis theory we can then adaptively change the model by assigning a value of $b_j \neq 0$ and T_j provided data is observed for $t > T_j$.

Thus, the variable order filter as applied to linear dynamic systems can be thought of as a curve-fitting approach to the solution of the problem of divergence of recursive filters due to model inaccuracies. This is accomplished in a method directly analogous to increasing the degree of polynomial in a curve fit to reduce the truncation error. The curve fitting can be done adaptively reducing the a priori knowledge required of the underlying signal function.

DD FORM 1 NOV 65 1473

UNCLASSIFIED
Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Variable order filter						
Constant order filter						
Digital adaptive filter						
Polynomial spline function						
Fixed-knot spline filter						
Variable-knot spline filter						

II

PAPER P-835

VARIABLE ORDER FILTERS

Marvin Blum

December 1971

INSTITUTE FOR DEFENSE ANALYSES
SCIENCE AND TECHNOLOGY DIVISION
400 Army-Navy Drive, Arlington, Virginia 22202

IDA Independent Research Program

CONTENTS

Preceding page blank

1. INTRODUCTION

Polynomial spline functions are a class of piecewise continuous polynomial functions satisfying continuity conditions at certain joint points, known as knots. They are a generalization of polynomials and have been found to have highly desirable characteristics as approximating, interpolating and curve-fitting functions. The latter property is of special interest in application to the problem of real-time estimation of state vectors of linear dynamic systems subject to intermittent discontinuous changes. Briefly stated, a polynomial spline function of order M is the M^{th} order integral of a step function. The points of discontinuity of the step function are the knots.

The theory of variable order filters was derived from the author's previous studies[7,11,1] on the application of polynomial splines and generalized splines[1,2]* to statistical filter theory. The current algorithms for the variable order filters are applications of and extensions of the spline function concepts. This class of filters will have application to smoothing and prediction of sampled data systems for the following classes of state estimation problems:

(a) Estimation of the state vector of a linear dynamic system with rapidly changing trajectories as, for example, a maneuvering vehicle;

(b) High precision estimation of the state vector based on long observational intervals where the estimation accuracy may be limited by modelling errors;

(c) Estimation of the state vector when certain components of the state vector are subject to intermittent discontinuous changes as, for example, staging rockets.

---

*Generalized splines are solutions to certain classes of homogeneous linear differential equations with suitably restricted time varying coefficients so as to ensure that the adjoint equation exists

Spline functions have been studied extensively, and two excellent books[1,2] present a comprehensive review of the applications and theory of spline functions.

C. deBoor and J. R. Rice[3,4] at Purdue University have programmed least-square cubic spline approximations for fixed-knot and variable-knot cases. In the fixed-knot case, the positions of the knots are fixed a priori; in the variable-knot case, the knot location is adjusted to the data set to minimize the mean-square error between the spline curve and the data set.

The property of splines which is of interest in this application is that for the same number of free parameters, the splines are able to more closely approximate a large class of functions than a polynomial curve fit. For example, if a cubic spline is used with two interior joint points, then each polynomial section has four coefficients to define the polynomial for a total of 12 coefficients. Typically, in the spline application, one requires the 0^{th} to 2nd derivative to be continuous at each interior knot point. This induces six linear equations of constraint on the coefficients, leaving six free parameters to be determined by least-squares curve fitting to the data set. The experience of previous investigators is that for some rather rapidly changing functions, such a spline function will follow the function more closely than, for example, a least-square polynomial of degree five, which also has six free parameters.

This has immediate implications for tracking, for example, maneuvering targets, since spline functions may be superior to polynomials as the basic approximation to the trajectory.

Tracking is essentially a real-time process, and the work of deBoor & Rice is essentially a post-flight analysis tool. That is, the algorithms are not suitable for real-time application. Real-time tracking using spline approximations was studed by H. Schneider and G. S. Gorden at MIT, 1966 - 1969[5,6]. In this application, Schneider considers the approximation of long segments of a ballistic reentry trajectory by a sequence of M^{th}-degree-polynomial splines with equally spaced knots. The first (M-1) derivatives were continuous at the interior knots. Schneider derives recursive algorithms in which the free parameters defining the splines are updated upon receiving a new observation in an optimal manner. That is, storing only the present observation and the previous estimate of the free parameter vector, Schneider obtains a minimum-variance estimate of the free parameters which is statisically equivalent to having processed all the observations in a batch process. Schneider also claims that the splines give a better fit (smaller bias error) than polynomials for the same number of free parameters.

In Reference (7) "Recursive Algorithms for Spline Filters," the author greatly expanded the model of the spline filter concept, to include linear combinations of known functions between arbitrarily spaced knots, and constraint conditions consisting of general linear operators evaluated at the knots. These generalizations are highly useful for the tracking of the class of trajectories of interest.

If dynamic equations of motion are available for the trajectory, one may linearize these equations about some nominal parameters which determine the trajectory. Then the generalized spline filters can be thought of as a

generalization of the Kalman filters as applied to piecewise-continuous functions, with additional linear constraints at the knot points. In this form, the adaptive techniques for placing knots may be an attractive method of avoiding some of the difficulties encountered in applying Kalman filters, especially divergence of the filter from the data. A special class of linear dynamic equations of motion with impulsive driving functions as the inputs at the knots form the basis for the variable order filter algorithms developed in Section 2.

In this paper, the concept of "variable order filters" has been applied to a trajectory given by a linear dynamic model with an impulsive driving function. The estimate of the state vector is obtained as a set of recursive equations as shown in Section 2. The time at which the impulsive driving function changes plays the role of the knots. The magnitude of the vector coefficients of the driving function are the unknowns to be estimated. In the non-adaptive mode the position of the knots are assumed, a priori. In the adaptive mode, the position of the knot is to be determined from the observations in real time.

The properties of the optimum variable order filter are:

A. It has a growing memory and uses all the data from the time origin to the present to obtain a minimum variance estimate of the state vector.

B. The state vector changes in dimension each time a knot is crossed when observing data in real time.

C. For sufficiently slow increase in bias error, there are possibilities for obtaining asymptotically consistent estimates of the state vector; that is, the total error including bias is driven to zero.

There is a trade-off in mean-square error for the variable order filter between the increase in dimensions of the state vector which increases the noise component of total error and the decrease in the bias error component. If the noise error increases too rapidly, it may not be desirable to increase the dimension of the state vector. An algorithm for collapsing to the dimension necessary to specify the trajectory between knots is shown also in Section 2.

The main computational penalty in using the optimum variable order filter seems to be storage. If the state vector grows to dimension ϕ the storage of the covariance matrix of the state vector is of the order $\phi^2/2$.

In Section 2 the algorithms for the variable order filters are derived for linear dynamic systems and presented in detail. This is the non-adaptive mode. In Appendix A a discussion of the adaptive version is given.

In Appendix A the algorithms developed in Section 1 are applied to the specific case of cubic splines. The polynomial splines and generalized splines form a subset of the class of functions which can be used to define the linear dynamic systems considered in this paper.

2. VARIABLE ORDER FILTERS FOR LINEAR DYNAMIC SYSTEMS

Consider the following linear dynamic system (Ref. 8):

$$\frac{dx}{dt} = A(t)x + f(t) \qquad t \in I \overset{\Delta}{=} [o, T) \tag{1}$$

$$y = M(t)x + \eta(t) \tag{2}$$

where $A(t) = [a_{ik}(t)]$ is an n x n matrix of functions, each of which is measurable on I and $|A(t)| \leq m(t)$, $t \in I$, where m(t) is Lebesgue-integrable on I. Similar properties hold for the elements of nxl column matrix f(t) and the rxn matrix M(t). Then it is shown in Ref. 9 that unique solutions exist for Eq. 1.

The solution can be written in terms of the initial state vector

$$x(o) \overset{\Delta}{=} x_o \text{ as}$$

$$x(t) = w_+(t, o)x_o + \int_o^t w_+(t, \lambda)\, f(\lambda)\, d \qquad t \geq o \tag{3}$$

Note that $w_+(t, \lambda)$ is the n x n state transition matrix determined for $t \geq \lambda$ as a solution of

$$\frac{dw_+(t, \lambda)}{dt} = A(t)w_+(t, \lambda) \qquad t > \lambda \tag{4}$$

with the initial condition

$$w_+(\lambda,\lambda) = I_n \tag{5}$$

where I_u is u x u identity matrix.

Note that $w_+(t, \lambda)$ is the response to a set of asymmetrical unit impulses $f_i(t) = \delta_+(t - \lambda)$ where the unit impulse $\delta_+(t - \lambda)$ is defined by

$$\int_{a+o}^{b} f(\varepsilon)\ \delta_+(\xi - x)\ d\xi = \begin{Bmatrix} 0 \text{ if } x < a \text{ or } \geq b \\ f(x + 0) \text{ if } a \leq x < b \end{Bmatrix} \quad a < b \tag{6}$$

the asymmetrical unit step function is given by

$$U_+(x) = \begin{matrix} 0 \\ 1 \end{matrix} \begin{Bmatrix} \text{for } x \leq 0 \\ \text{for } x > 0 \end{Bmatrix} \tag{7}$$

In Eq. 2, y is an r x 1 observation vector M(t) is an n x r observation system transformation with continuous elements $M_{ij}(t)$ and $\eta(t)$ is an r x 1 vector representing white observation noise. The relationship between the observation and the initial state vector can be written by substituting Eq. 3 into Eq. 2 as

$$y = M(t)w_+(t,o)x_o + M(t) \int_o^t w_+(t, \lambda)f(\lambda)\ d\lambda + \eta(t) \qquad t > o \tag{8}$$

The forcing function f(t) will be assumed to be a sum of impulses. This will have the effect of discontinously changing the state vector immediately after a set of time values T_i, $i = 1, 2, \ldots N$ which are known as the knots. Let

$$f(t) \triangleq \sum_{i=1}^{N} \delta_+(t - T_i)b_i \tag{9}$$

where the b_i are unknown components of an augmented state vector. Then

$$y(t) = M(t)w_+(t, o)x_o + M(t) \sum_{i=1}^{N} w_+(t, T_i)b_i + \eta(t) \tag{10}$$

Note that

$$w_+(t, T_i) = o \qquad t < T_i \tag{11}$$

Thus, as time progresses, we obtain in a natural way a variable order filter as follows: Let $o \leq t \leq t_1$ then

$$y = M(t)w_+(t,o)x_o + \eta(t) \tag{12}$$

for which the state vector can be defined as

$$x^{[o]}(t) = w_+^{[o]}(t, \tau)x(\tau) \tag{13}$$

and is of dimension n. Let $o \leq t \leq T_2$, then

$$y + M(t)\ [w_+(t,\ o)x_o + w_+(t,\ T_1)b_1] + \eta(t) \tag{14}$$

Define an augmented state vector as

$$x^{[1]'}(t) = \begin{vmatrix} x^{[o]}(t) \\ b_1 \end{vmatrix} \qquad T_2 \geq t \geq o \tag{15}$$

with transition matrix $w_+^{[1]'}(t,\ o)$ given by

$$w_+^{[1]'} = \begin{vmatrix} w_+(t,\ o) & w_+(t,\ T_1) \\ o & U_+(t-T_1)I_n \end{vmatrix} \tag{16}$$

Therefore

$$x^{[1]'}(t) = w_+^{[1]'}(t,\ o)\ x^{[1]'}(o) \tag{17}$$

The state vector has increased in dimensions from n to 2n. In general it is not desirable to increase the state vector dimensions any more than necessary. Let it be known a priori that only some subset $q_{(j)}$ of components of the n x 1 vectors b_j will differ from zero. Usually this will be the last component b_{jn}. In this case, if we let b_j^* represent the $q_{(j)}$ x 1

vector of nonzero components (rows) and $w_+^*(t, T_j)$ the corresponding columns of $w_+(t, T_j)$. Then we may write Eq. 16 in terms of the nonzero components of b_j^* and the state vector $x^{[o]}(\tau)$ as

$$\begin{vmatrix} x^{[o]}(t) \\ \\ b_1^* \end{vmatrix} = \begin{vmatrix} w_+(t, \tau) & w_+^*(t, T_1) \\ \\ o & U_+(t - T_1)I_{q(1)} \end{vmatrix} \begin{vmatrix} x^{[o]}(\tau) \\ \\ b_1^* \end{vmatrix} \tag{18}$$

or

$$x^{[1]}(t) = w_+^{[1]}(t, \tau) \quad x^{[1]}(\tau) \tag{19}$$

Note the dropping of the ' notation for $x^{[1]}(t)$, $w_+^{[\ ]}(t, \tau)$ indicates that b* is being used.

Let y(t) be observed over the discrete set of times $(t_1, t_2, t_3 \ldots t_{k+1})$. The following section presents the recursive algorithms for the minimum variance estimates of the state vector at the current time t_{k+1} and the equations for the covariance matrix of the state vector estimation. The minimum variance estimate corresponds to a growing memory filter operating on all the data from the first measurement t_1 to the current time t_{k+1}. The state vector grows in dimension by q(u) at the beginning of the U^{th} stage or when data at time $t > T_u$ (the u^{th} knot) is acquired. In Section 3.2 the algorithms for a constant order filter are described. In this filter, the state vector is updated at the beginning of the u^{th} stage, but maintained

at the dimension n. This algorithm is similar to a constant memory filter and is not optimal in a minimum variance sense.

The constant order filter, however, requires less storage and under some circumstances, may be more desirable if the noise error due to increased dimension of the state vector is growing more rapidly than the bias or dynamic modelling error.

In the algorithms of Sections 3.1 and 3.2, it is assumed that the positions of the knots are known a priori. Possible modifications of the algorithms to make them adaptive, i.e., to determine T_j from the data, are discussed in Appendix B.

3. RECURSIVE ALGORITHMS FOR VARIABLE ORDER FILTER

In general, we can write the following recursive relationship for the state vectors, transition matrices and observation matrix as a measurement in a new stage is introduced. Define for U = 0:

$$x^{[o]}(t) \triangleq x(t)$$

$$T_1 \geq t \geq \tau \geq o \equiv T_o$$

$$w_+^{[o]}(t, \tau) \triangleq w_+(t, \tau) \tag{20}$$

$$M^{[o]}(t) \triangleq M(t)$$

$$b_o^* = 0$$

Define for U = 0, 1, 2, ... N - 1, and for $T_{(U+1)} \geqq t > T_U$

$$x^{[U+1]}(t) = \begin{vmatrix} x^{[U]}(t) \\ b^*_{U+1} \end{vmatrix} \tag{21}$$

$$w_+^{[U+1]}(t,\tau) = \left|\begin{array}{c|c} w_+^{[U]}(t,\tau) & \begin{array}{c} w_+^*(t, T_{U+1}) \\ \hline 0 \end{array} \\ \hline 0 & U_+(t-T_{U+1})I_{q(U+1)} \end{array}\right| \begin{array}{c} \sum_{j=1}^{U} q(j) \end{array} \tag{22}$$

$$M^{[U+1]}(t) = [M^{[U]}(t) \mid 0] \tag{23}$$

Let $y(t)$ be observed over a discrete set of observation times $\{t_1, t_2, \ldots t_k\}$.

Let

a. $\hat{x}^{[U]}(k|k)$ be the minimum variance state estimat at t_k given all the observations up to and including t_k.

b. $P^{[U]}(k|k)$ be the covariance matrix of the error in $\hat{x}^{[U]}(k|k)$.

c. $\psi_{+}^{[U]}(k + 1,k)$ be the state transition matrix from t_k to t_{k+1}.

d. $\hat{x}^{[U]}(k + 1|k)$ be the minimum variance estimate of the predicted state vector at time t_{k+1} based on the measurements $Y_k:\{y(t_1), y(t_2) \ldots y(t_k)\}$.

e. $P^{[U]}(k + 1|k)$ be the covariance matrix of the error in $\hat{x}^{[U]}(k + 1|k)$.

f. $M^{[U]}(k + 1)$ be the augmented measurement matrix at t_k.

g. $R(k + 1)$ be the measurement noise covariance matrix associated with $\eta(t)$ at t_k.

h. $K^{[U]}(k + 1)$ be the filter gain at t_{k+1} and y_{k+1} be the measurement or observation at t_{k+1}.

Assume that the initial conditions x_o are random variables with ensemble mean $\bar{x}_o$ and covariance matrix P_o known a priori. For the b_j^* assume a priori estimates of $\overline{b_j^*}$ and its corresponding covariance matrices are known. Let x_o, b_j^* be independent of each other and the measurement noise as well as the position of the knots $T_1, \ldots T_N$. Then one may write the following algorithm (Ref. 10, page 270).

3.1 ALGORITHM A

Within the U^{th} stage for samples $T_{U+1} \geq t_k > T_U$

Suppose: $t_k > T_U$, $t_{k-1} \leq T_U$: $U = 1, \ldots N - 1$

Initialize the U^{th} stage by defining the a priori state vector:

$$1'.\quad \hat{x}^{[U]}(k-1|k-1) \triangleq \left| \begin{array}{c} \hat{x}^{[U]}(k-1|k-1) \\ \\ \overline{b_U^*} \end{array} \right|$$

The a priori covariance vector is given by

$$2'.\quad P^{[U]}(k-1|k-1) \triangleq \left| \begin{array}{c|c} P^{[U]}[k-1|k-1) & o \\ \hline o & \mathrm{Var}\,\{b_U^*\} \end{array} \right|$$

Form the augmented transition matrix:

$$3'.\quad w_+^{[U]}(t_k, t_{k-1}) = \begin{array}{|c|c|l} & w_+^*(t_k, T_U) & n \\ \cline{2-2} w_+^{[U-1]}(t_k, t_{k-1}) & 0 & \sum_{j=1}^{U-1} q(j) \\ \hline 0 & U_+(t_k - T_U) I_{q(u)} & q(U) \end{array}$$

Form the augmented observation matrix:

$$4'.\ M^{[U]}(t_k) = [M^{[U-1]}(t_k)|0]$$

Compute the predicted state vector:

$$5'.\ \hat{x}^{[U]}[k|k-1] = w_+^{[U]}[k, k-1]\ \hat{x}^{[U]}[k-1|k-1]$$

Compute the predicted error covariance:

$$6'.\ P^{[U]}[k|k-1] = w_+^{[U]}[k, k-1]\ P^{[U]}[k-1|k-1]\ w_+^{[U]}[k, k-1]^T$$

Compute the filter gain matrix:

$$7'.\ K^{[U]}(k) = P^{[U]}[k|k-1]M^{[U[}(k)^T[M^{[U]}(k)P^{[U]}(k|k-1)M^{[U]}(k)^T+R(k)]^{-1}$$

Process the observations y_k to update state vector

$$8'.\ \hat{x}^{[U]}(k|k) = \hat{x}^{[U]}(k|k-1) + K^{[U]}(k)[y_k-M^{[U]}(k)\hat{x}^{[U]}[k|k-1]]$$

Compute the covariance matrix

$$9'.\ P^{[U]}(k|k) = \left[I_{Q(U)} - K^{[U]}(k)\ M^{[U]}(k)\right] P^{[U]}(k|k-1)\ x \left[I_{Q(U)} - K^{[U]}(k)\ M^{[U]}(k)\right]^T + K^{[U]}(k)R(k)\ K^{[U]^T}$$

where

$$Q(U) = n + \sum_{j=1}^{U} q(j)$$

$$Q(o) = n$$

$Q(U)$ is the dimension of the state vector in the U^{th} stage.

Test:

$$10'.\ t_{K+1} \leq T_{U+1} \rightarrow \text{yes} \quad \text{set } k = k + 1, \text{ and return to Eq. 3.}$$

If no, set U = U + 1 and suppose j(U) is the smallest integer for which

$$t_{k+j(U)} > T_{U+1} \quad , \ t_{k+j(U)-1}$$

Set k + j(U) for k in Eq. 1' through Eq. 10' and start at Eq. 1'.

3.1.1 Initiation Procedure, Zeroth Stage

For U = 0 Eq. 1 through Eq. 10, reduce to the standard linear Kalman filter (Ref. 10, page 270) in which the state vector transition matrix and observation matrix are given by Eq. 20 valid for $T_1 \geq t \geq 0$. Eq. 5' becomes at t_1, that is, k = 1,

$$\hat{x}^{[o]}[1|o] = w_+[1,o]x_o$$

$$P^{[o]}[o|o] \triangleq P_o$$

Equations 6' through 10' are then obtained by obvious substitution. More generally, when one wishes to change from a variable order to a fixed order filter of dimension of the original state vector, one may derive initial state vectors and covariance matrices at time t_{k-1} noted by $\hat{\underline{x}}(k - 1|k - 1)$, $\underline{P}(k - 1|k - 1)$ and substitute M(k) for $M^{[U]}(k)$, $\underline{w}^{[U]}(k, k - 1)$ for $w^{[U]}(k, k - 1)$ in Eqs. 5' through 9'. The presence of the _ then indicates a constant order filter.

3.2 ALGORITHM B, ALGORITHM TO CONVERT TO CONSTANT ORDER FILTER

Suppose that at observation t_{k-1} in the U^{th} stage, it is decided to switch from a growing order filter to a constant order filter. Then assume one has computed from the observation at t_{k-2} both $\hat{x}^{[U]}(k-2|k-2)$ and $P^{[U]}(k-2|k-2)$. Let

3.* $\underline{w}^{[U]}(k-1,k-2)$ = first n rows of $w_{+}^{[U]}(k-1,k-2)$

1.* and $\underline{\hat{x}}^{[U]}(k-1|k-2)$ = first n elements of the state vector $\hat{x}^{[U]}(k-1|k-2)$

Then it follows that:

5.* $\underline{\hat{x}}^{[U]}(k-1|k-2) = \underline{w}^{[U]}(k-1,k-2)\ \hat{x}^{[U]}(k-2,k-2)$

and the covariance of $\underline{\hat{x}}^{[U]}(k-1|k-2)$ is given by

6.* $\underline{P}(k-1|k-2) = \underline{w}^{[U]}(k-1,k-2)P^{[U]}(k-2|k-2)\underline{w}^{[U]}(k-1,k-2)^T$

The gain matrix is given by Eq. 7' with

7.* $\underline{K}(k-1) = P((k-1|k-2))\underline{M}(K-1)^T\ [M(k-1)\underline{P}(k-1|k-2)M(k-1)^T+R(k-1)]^{-2}$

The updated state vector and covariance matrix after processing the observation at t_{k-1} is given by

9.* $$\underline{P}(k-1|k-1) = [I_n-\underline{K}(k-1)M(k-1)]\underline{P}(k-1\ k-2)[I_n-\underline{K}(k-1)M(k-1)]^T$$

$$+ \underline{K}(k-1)R(k)\underline{K}(k-1)^T\ .$$

---

*The starred equations are parallel to the same equations in Algorithm A.

$$8.^{*}\quad \hat{\underline{x}}(k-1|k-1) = \underline{x}^{[U]}(k-1|k-2)+\underline{K}(k-1)[y_{k-1}-M(k-1)\hat{\underline{x}}^{[U]}[k-1|k-2]]$$

Equations 8* and 9* of Algorithm B derive a state vector of dimensions n and its covariance matrix which contain the effects of each of the impulse changes to the state vector.

3.2.1 Approximately Constant Order Filter

Suppose a new knot appears and we do not wish to increase the dimensionality of the state vector. We can increase the dimensionality temporarily for a set of measurement to obtain an estimate of b_j^* and then collapse the order using Algorithm B. Let us demonstrate this for a specific example as follows: One starts in the zeroth stage with a state vector of dimension n. Suppose at each knot, the impulse input b_i has only the last component as a nonzero value with mean $\overline{b}_{in}$ and variance σ_i^2. In the zeroth stage one has a state vector of dimension n for the k - 1 observation. Note $t_{k-1} \leq T_1$, $t_k > T_1$, and U = 0, is used in Algorithm A. For the next j-measurements one uses a state vector of dimension n + 1 applying Algorithm A for U = 1 with $t_{k+j-1} \leq T_2$, $t_{k+j} > T_2$. Letting t_{k+j-1} replace t_{k-1} in Algorithm B, we obtain "initial" conditions at time t_{k+j-1}, that is, the n dimension state vector $\{\underline{x}(k+j-1|k+j-1)\}$, and its (n x n) covariance matrix $\underline{P}(k+j-1|k+j-1)$. We may now return to Algorithm A and augment $\hat{\underline{x}}(k+j-1|k+j-1)$ by adding $\overline{b}_{2n}$, in Eq. 1' and augment σ_2^2 to the a priori covariance matrix in Eq. 2'.

The effect of collapsing the dimension of the state vector is to obtain a suboptimal estimate of the state vector in the sense that it is not a minimum variance estimator. However, this may be desirable to control the

total error or divergence of the model from the data, and reduce storage requirements.

REFERENCES

(1) "Theory and Applications of Spline Functions," edited by T. N. Greville, Academic Press, London, 1969.

(2) "The Theory of Splines and Their Applications," Nilson, Ahlberg, Walsh, Academic Press, N.Y. and London, 1967.

(3) deBoor, C. and Rice, J. R., "Lease Square Cubic Spline Approximation I: Fixed Knots," Report CSD TR20, Computer Science Department, Purdue University, 1968.

(4) deBoor, C. and Rice, J. R., "Lease Square Cubic Spline Approximation II: Variable Knots," Report CSD TR21, Computer Science Department, Purdue University, 1968.

(5) Schneider, H., "Matched Polynomial Lease Square Fitting to Real-Time Ballistic Coefficient Estimation," Tech Note 1966-34, Lincoln Laboratories, M.I.T., 1966, DDC-AD-637669.

(6) Schneider, H. and Gorden, G. S., "Real-Time Trajectory Estimation Using Spline Approximations," M.I.T. Labs., TN-1969-37, 1969.

(7) "Recursive Algorithms for Spline Filters," Marvin Blum, 21 January 1971, System Development Corporation SP-3583. Presented at Second Symposium on Non-Linear Estimation Theory and Its Applications, Sept. 13-15, 1971, San Diego, California, and available in the Proceedings of the Symposium (Sponsored by IEEE Control System Society).

(8) G. H. Korn and T. M. Korn, "Mathematical Handbook for Scientist and Engineers," 2nd Edition, McGraw-Hill Book Co., N.Y., 1968, p. 423.

(9) E. A. Coddington and N. Levenson, "Theory of Ordinary Differential Equations, McGraw-Hill, Inc., N.Y., 1955.

(10) Andrew J. Jazwinski, "Stochastic Processes in Filter Theory," Academic Press, New York, 1970.

(11) "Recursive Spline Filters with Correlated Errors," Marvin Blum, March 1971, System Development Corporation SP-3583/001/00, Santa Monica, California.

(12) "On the Optimal Smoothing of Piecewise Continuous Functions," Marvin Blum, IEEE Transactions on Information Theory, March 1972.

APPENDIX A

EXAMPLE OF ALGORITHMS A AND B

EXAMPLE CUBIC SPLINES

Let $$x(t) = x_o + \dot{x}_o\, t + \ddot{x}_o\, t^2/2! + \dddot{x}_o\, t^3/3! \quad t \geq 0 \qquad \text{(1A)}$$

where the initial state vector is

$$x_o^T = \{x_o, \dot{x}_o, \ddot{x}_o, \dddot{x}_o\}$$

position (x_o), velocity ($\dot{x}_o$) acceleration ($\ddot{x}_o$) and jerk ($\dddot{x}_o$) at t=o.

Then it follows that $\dfrac{d^4x(t)}{dt} = 0$

From the Eq. 3 with $f(t) \equiv 0$ one has the homogeneous state equation

$$\dot{x}(t) = w_+ (t,\tau)\, x\, (\tau) \qquad \text{(2A)}$$

By the Taylors expansion of (1A) about τ one can easily show that

$$w_+(t,\tau) = \begin{bmatrix} 1 & (t-\tau) & \dfrac{(t-\tau)^2}{2!} & \dfrac{(t-\tau)^3}{3!} \\ 0 & 1 & (t-\tau) & \dfrac{(t-\tau)^2}{2!} \\ 0 & 0 & 1 & (t-\tau) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \underline{t \geq \tau \geq 0} \qquad \text{(3A)}$$

Let the samples occur at integral values of $t = 1, 2, \ldots$ and the knots occur at $T_j = N j$. $j = 1, 2, 3, \ldots$, with $N = 10$.

Further let the input driving function $f(t)$ be given by

$$f(t) = \sum_{j=1}^{\infty} b_j \, \delta_+(t - T_j) \qquad (4A)$$

Where the elements of the nx1 vector b_j are known to be zero except for the n^{th} element b_{nj} and

$$E\{b_{nj}\} = 0$$

$$\text{Var}\{b_{nj}\} = \sigma_j^2$$

The a priori values of $E\{x_o\} = \bar{x}_o = 0$ and Var $\{x_o\} = P_o$ are known. The random variables b_{nj}, x_o and the measurement error are mutually independent. The observations are given by the scalar

$$y(t) = x(t) + \eta(t) \qquad (6A)$$

where

$$E\{\eta(t)\} = 0, \; E\{\eta(k)\eta(j)\} = \delta_{jk} R^2(k) \qquad (7A)$$

$$\delta_{jk} = 1 \quad j = k, \text{ otherwise } 0.$$

That is $M(t) = [1,0,0, \ldots 0]$. The recursive algorithm A is used for $t = 1, 2, \ldots N$

$$\hat{x}^{[o]}[k|k-1] = w_{+}^{[o]}[k,k-1]\,\hat{x}^{[o]}[k-1|k-1] \tag{8A}$$

where

$$\hat{x}^{[o]}[0|0] = \bar{x}_o = 0$$

$$P^{[o]}[k|k-1] = w_{+}^{[o]}[k,k-1]\,P^{[o]}[k-1|k-1]\,w_{+}^{[o]}[k,k-1]^T \tag{9A}$$

with

$$P^{[o]}[k-1|k-1] = P_o,\ k=1 \tag{10A}$$

$$K^{[o]}(k) = P^{[o]}[k|k-1]\,e_1^{n^T}\left[P_{11}^{[o]}[k|k-1] + R^2(k)\right]^{-1}$$

where

$e_j^n = [0, 0, \ldots 1, 0, \ldots 0]$ is the 1xn unit vector with zeros for each component except the j^{th} which is 1. The double subscript on P indicates the element of the matrix. The optimum estimate of the current value of the state vector is then

$$\hat{x}^{[o]}[k|k] = \hat{x}^{[o]}[k|k-1] + K^{[o]}(k)[y_k - \hat{x}_1^{[o]}[k|k-1]] \tag{11A}$$

with covariance matrix [see alternate form page 270 Ref. 10].

$$P^{[o]}(k|k) = [I_n - K^{[o]}(k)\ e_1^n]\ P^{[o]}(k|k-1] \tag{12A}$$

when k = 10, and impulse input occurs which effects the filter at k = 11. We now set U = 1 and switch back to Eq. 1' of Algorithm A. Setting k = 11 for the initial condition of U = 1 the a priori state vector is given by

$$x^{[1]}(10,10) = \begin{bmatrix} \hat{x}^{[o]}[10,10] \\ \\ \bar{b}_{1n} = 0 \end{bmatrix} \tag{13A}$$

The a priori covariance matrix is given by

$$P^{[1]}(10\ 10) = \left|\begin{array}{c|c} P^{[o]}[10|10] & 0 \\ \hline 0 & \sigma_1^2 \end{array}\right| \tag{14A}$$

The augmented transition matrix is given by

$$w_+{}^{[1]}(k,k-1) = \left|\begin{array}{c|c} & \frac{1}{3!}(k-10)_+^3 \\ w_+(k,k-1) \text{ from Eq. 3} & \frac{1}{2!}(k-10)^2{}_+ \\ & (k-10)_+ \\ & U_+(k-10) \\ \hline 0 & U_+(k-10) \end{array}\right| \tag{15A}$$

for $\quad k = 11,12,\ldots$

where $(U)_+ = 0 \quad U \leq 0$

$= 1 \quad U > 0$

$$M^{[1]}(k) = e_1^{n+1} \tag{16A}$$

$(U)_+^j = U^j,\ U > 0$

$= 0 \quad U \leq 0$

The recursive formulation given by Eq. 5' to 10' are then followed with obvious modifications as shown in equations A-8 to A-12 until k = 20. At k = 20, a second impulse enters which requires changing U from 1 to 2 and looping back to equation 1' of algorithm A as follows:

$$\hat{x}^{[2]}[20|20] = \begin{bmatrix} \hat{x}^{[1]}[20|20] \\ \\ \bar{b}_{2n} = 0 \end{bmatrix} \tag{17A}$$

$$P^{[2]}[20|20] = \left[\begin{array}{c|c} P^{[1]}[20|20] & 0 \\ \hline 0 & \sigma_2^2 \end{array}\right] \tag{18A}$$

$$w_+^{[2]}(k,k-1) = \left|\begin{array}{c|c} & \frac{1}{3!}(k-20)_+^3 \\ w_+^{[1]}(k,k-1) & \frac{1}{2!}(k-20)_+^2 \\ & (k-20)_+ \\ & U_+(k-20) \\ \cline{2-2} & 0 \\ \hline 0 & U_+(k-20) \end{array}\right| \quad k = 21, 22, \ldots \tag{19A}$$

$$w_+^{[2]}(k,k-1) = \left|\begin{array}{c|c|c} & (k-10)_+^3/3! & (k-20)_+^3/3! \\ & (k-10)_+^2/2! & (k-20)_+^2/2! \\ W_+(k,k-1) & (k-10)_+ & (k-20)_+ \\ & U_+(k-10) & U_+(k-20) \\ \hdashline 0 & U_+(k-10) & 0 \\ 0 & 0 & U_+(k-20) \end{array}\right| \tag{19B}$$

$$M(k) = e_1^{n+2} \tag{20A}$$

for k = 21, 22, ...

Algorithm A is now used with obvious substitutions for k=21, ... 30

EXAMPLE OF CONSTANT ORDER FILTER

Let us use the sample example of the cubic splines to demonstrate the constant order filter algorithm B. In the zeroth stage one has obtained

the optimal estimate of the 1x4 state vector x. To accommodate the family of impulse responses consisting of 1 non-zero component, the state vector is augmented as given by Eq. A-13, to a 1x5 state vector and Algorithm A is exercised. In the third stage instead of augmenting the state vector as shown by Eq. A-17, it is desired to maintain the state vector as a 1x5. This will be done using algorithm B as follows.

We wish to end up with initial conditions for the third stage that is k-1=20 in Equations 8',9' Algorithm B. Therefore, take k=21 and write

$$\underline{w}^{[1]}(20,19) = \left|\begin{array}{cccc|c} 1 & 1 & \frac{1}{2!} & \frac{1}{3!} & \frac{1}{3!}(20-10)^3 \\ 0 & 1 & 1 & \frac{1}{2!} & \frac{1}{3!}(10)^2 \\ 0 & 0 & 1 & 1 & 10 \\ 0 & 0 & 0 & 1 & 1 \end{array}\right| \tag{21A}$$

$$\hat{x}^{[1]} = \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \hat{x}_4 \\ \hat{b}_{1n} \end{bmatrix} \tag{22A}$$

$$\underline{x}^{[1]}[20|19] = \underline{w}^{[1]}(20,19)\ \hat{x}^{[1]}[19|19]\ . \tag{23A}$$

Note $x^{[1]}[19|19]$ is a 1x5 state vector and $P^{[1]}[19|19]$ is a 5x5 covariance matrix of $x^{[1]}$. They are given from algorithm A or previous processing. Then

$$\underline{P}(20|19) = \underline{w}^{[1]}(20,19)\, P^{[1]}[19|19]\, \underline{w}^{[1]}(20,19)^T \tag{24A}$$

$$= \text{4x4 covariance matrix}$$

$$\underline{K}(20) = \underline{P}(20|19)\, e_1^4\, [\underline{P}_{11}(20|19) + R^2(20)]^{-1} \tag{25A}$$

$$\underline{P}(20|20) = [I_4-\underline{K}(20)e_1^4]\, \underline{P}(20|19)\, [I_4-\underline{K}(20)e_1^4]^T \tag{26A}$$

$$+ \underline{K}(20)\, R\, (20)\, \underline{K}(20)^T$$

$$\underline{x}(20|20) = \underline{x}^{[1]}[20|19] + \underline{K}(20)\left\{ y_{20} - x_1{}^{[1]}[20|19] \right\} \tag{27A}$$

Note that the ^ is deleted from the estimate $\underline{x}$ since once the collapse form of the state vector is used, it is no longer a minimum variance estimate.

We have now obtained initial conditions given by equations 26A and 27A in terms of the 4x1 state vector. In the next stage (U=3) we augment the state vector to accommodate the impulse input b_{2n} and use algorithm A beginning with equation 1? in the modified form shown in Equation 22

$$w_{+}^{[2]}(k,k-1)^{*} \triangleq \left|\begin{array}{cccc|cc} 1 & 1 & \frac{1}{2!} & \frac{1}{3!} & \frac{1}{3!} & (k-20)_{+}^{3} \\ 0 & 1 & 1 & \frac{1}{2!} & \frac{1}{2!} & (k-20)_{+}^{2} \\ 0 & 0 & 1 & 1 & \frac{1}{1!} & (k-20)_{+} \\ 0 & 0 & 0 & 1 & U_{+} & (k-20)_{+} \\ \hline 0 & 0 & 0 & 0 & U_{+} & (k-20) \end{array}\right| \tag{28A}$$

$$M^{[2]}(k) = e_1^5 \tag{29A}$$

$$P^{[2]}[20,20]^{*} \triangleq \left[\begin{array}{c|c} \underline{P}(20,20) & 0 \\ \hline 0 & \sigma_2^2 \end{array}\right] \tag{30A}$$

$$x^{*[2]}(20,20) = \begin{bmatrix} \underline{x}(20,20) \\ E\{b_{2n}\} = 0 \end{bmatrix} \tag{31A}$$

Equations 28A to 31A are now the input to algorithm A starting with equation 5' and k=21. Note that at k=30 when the next impulse input occurs one may loop back to equation 1' of algorithm A and let the order of the filter grow or collapse the state vector to dimension 1x4 at observation k=30 keeping the state vector at dimension 5 during the next stage. We

note that once the order of the filter has been reduced, the state estimates are no longer minimum variance. They are still unbiased. This is the price to be paid to save the storage and processing of the larger dimension state vector.

APPENDIX B

ADAPTIVE VARIABLE ORDER FILTERS

The formalism of the algorithms as presented in this paper are sufficiently simple and compact to make possible:

(a) Adaptive placement of knots based on measured bias error.

(b) Search procedures to determine the existence of a staging or maneuver and modification of the last knot to accommodate a detected maneuver.

The techniques for solving (a) are readily extended to the application described by (b) by requiring more elaborate detection schemes. In the maneuvering case one may wish to determine in steps of increasing complexity:

(a) If a state vector change has caused a build up of bias error;

(b) In what interval and how large a change occurred;

(c) At what sample and how large a change occurred;

(d) At what point in time and how large a change occurred.

The structure of the Adaptive Variable Order Filter is shown in Figure 1. The observations $\{y_1 \ldots y_k\}$ are the input to the variable order filter from which a predicted state vector $\hat{x}(k+1|k)$ is obtained. The predicted observation is compared with the actual observation y_{k+1} and the residual $\{y_{k+1} - \hat{y}(k+1|k)\}$ is used as the input to the bias detector. The form of the bias detector will vary with the application. The detector may also operate on a block of s residuals rather than just 1. The output of the bias detector will be, an estimate of the magnitude of the bias error, and

possibly positional information for knot point determination. Based on this output a decision is made as to whether or not to change the structure of the filter. If a change is decided, one may place an additional knot, increasing the dimension of the state vector. This procedure would be followed if continued improvement in accuracy is attainable and desired. Alternately one may place a knot but not increase the dimensions of the state vector. This would be equivalent to fixing the memory length of the filter. The latter procedure will require less computer storage and may be desirable if maximum accuracy is not required or improved accuracy is not obtainable.

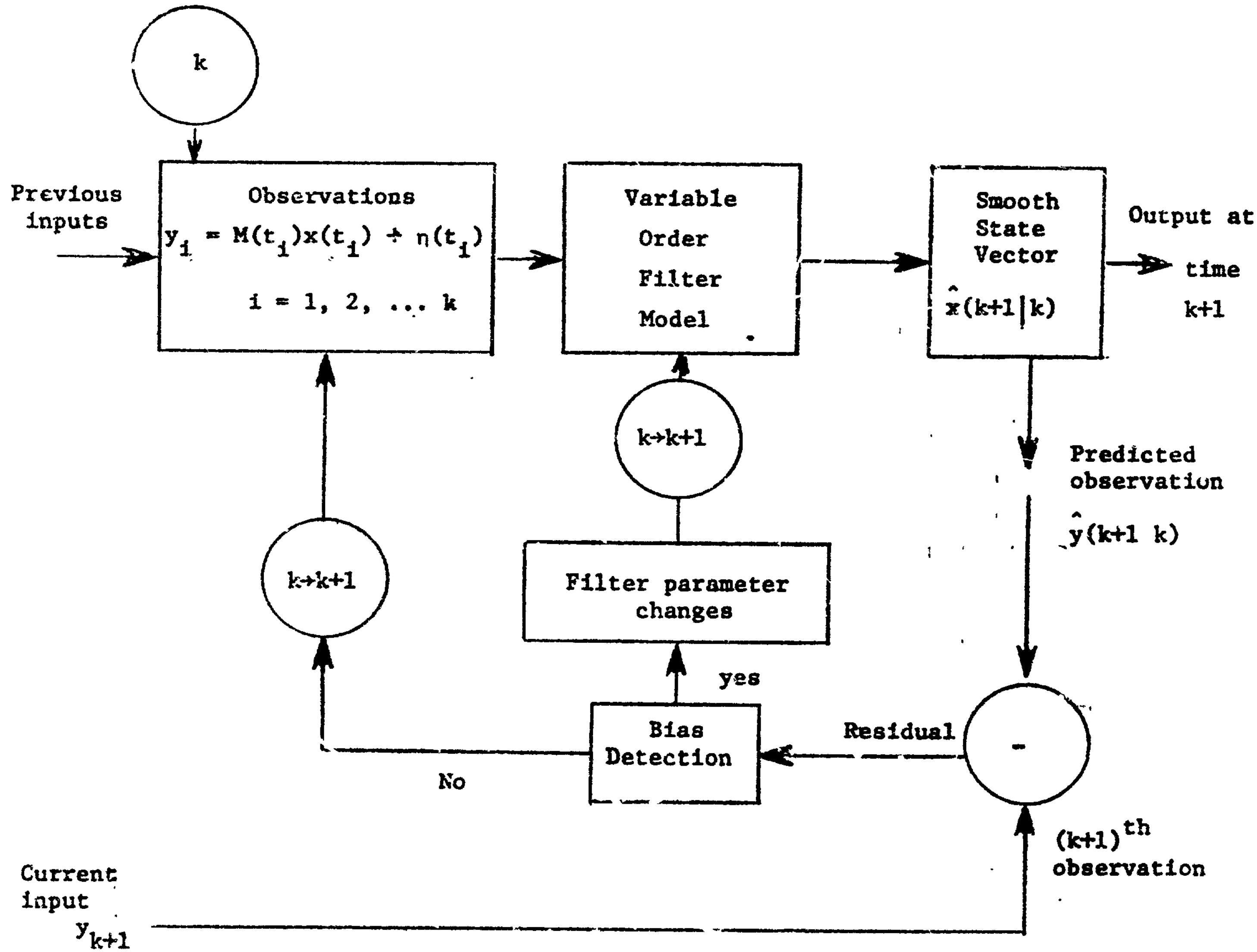

Figure 1 STRUCTURE OF THE ADAPTIVE VARIABLE ORDER FILTER